



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/955,764	09/19/2001	Jun Li	10007965	9833

7590 06/07/2004

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, CO 80527-2400

EXAMINER

RUTTEN, JAMES D

ART UNIT	PAPER NUMBER
2122	

DATE MAILED: 06/07/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/955,764	LI ET AL.
Examiner	Art Unit	
J. Derek Ruttan	2122	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) Responsive to communication(s) filed on 19 September 2001.

2a) This action is **FINAL**. 2b) This action is non-final.

3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) Claim(s) 1-43 is/are pending in the application.
4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5) Claim(s) _____ is/are allowed.
6) Claim(s) 1-43 is/are rejected.
7) Claim(s) _____ is/are objected to.
8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) The specification is objected to by the Examiner.

10) The drawing(s) filed on 19 September 2001 is/are: a) accepted or b) objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) All b) Some * c) None of:
1. Certified copies of the priority documents have been received.
2. Certified copies of the priority documents have been received in Application No. _____.
3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) Notice of References Cited (PTO-892)
2) Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date 2, 3.
4) Interview Summary (PTO-413)
Paper No(s)/Mail Date. ____.
5) Notice of Informal Patent Application (PTO-152)
6) Other: ____.

DETAILED ACTION

1. Claims 1-43 have been examined.

Drawings

2. The drawings are objected to under 37 CFR 1.83(a). The drawings must show every feature of the invention specified in the claims. Therefore, the following details must be shown or the feature(s) canceled from the claim(s):

- a. Instrumented stub/skeleton is generated from a description of an interface (claims 2, 10 and 11)
- b. Log data contents are configured during generation of stub (claim 6)
- c. Log data contents are configured during operation (claim 7)
- d. Runtime information includes a regular expression (claim 8)
- e. Log contents changeable by using regular expression (claim 8)
- f. Event number is updated before being copied into storage (claim 22)
- g. Log data comprises a self thread identifier for thread distinction and optionally a function container identifier (claim 24)
- h. Intercepting memory requests and logging heap memory usage (claim 25)

No new matter should be entered.

A proposed drawing correction or corrected drawings are required in reply to the Office action to avoid abandonment of the application. The objection to the drawings will not be held in abeyance.

Claim Rejections - 35 USC § 112

3. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

4. Claim 37 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

5. Claim 37 recites the limitation " said first and stub end log data " in line 2. There is insufficient antecedent basis for this limitation in the claim. This phrase has been interpreted to mean --said stub start and end log data--.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. Claims 1-7, 9-11, 13-19, 21-32, and 35-42 are rejected under 35 U.S.C. 103(a) as being unpatentable over prior art of record "JaViz: A client/server Java profiling tool" by Kazi et al. (hereinafter referred to as "Kazi") in view of "Automatic Insertion of Performance Instrumentation for Distributed Applications" by Blumson et al. (hereinafter referred to as "Blumson").

As per claim 1, Kazi discloses:

A monitoring method for a component-based software system operating over one or more processing devices (Kazi page 1, Abstract: “The JaViz performance analysis tool generates **execution traces** with sufficient detail to determine program hot spots, **including remote method calls**, in a **distributed Java application program**”; also page 8 paragraph 3: “...executing on a physically distributed **processor**.”), *comprising the steps of*:

initiating an invocation of a second software component from within an execution of a first software component (Kazi page 8 paragraph 3 under “Client/server trace generation”: “The Java remote method invocation (RMI) facility allows **one Jvm to execute a method on another Jvm**, which may be executing on a physically distributed processor.”);

recording a stub start log data before said invocation of said second software component (Kazi page 7 last paragraph under “Detailed trace generation”: “The trace generation module of **the Jvm is modified to record every invocation** of a method using time stamps that **show the start and end times** of the method with microsecond resolution”);

recording a stub end log data in said instrumented stub after a response is received from said invocation of said second software component (Kazi page 7 last paragraph as cited above);

wherein said stub start log data and said stub end log data gather runtime information about execution of said second software component within said component-based software system (Kazi page 7 last paragraph: “Additionally, a thread identifier is recorded to uniquely identify the thread executing the method.”).

Kazi does not expressly disclose an instrumented stub.

However, in an analogous environment, Blumson teaches instrumenting a stub to collect runtime data (page 6, Section 6.1: “Our IDL compiler has an additional command-line flag...to **insert instrumentation**.”).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Blumson’s stub instrumentation implementation in Kazi’s instrumented jvm. One of ordinary skill would have been motivated to take measurements on certain operations such as marshalling time that are otherwise difficult, while maintaining a relatively simple implementation versus modification of a runtime library.

As per claim 2, the above rejection of claim 1 is incorporated. Kazi does not expressly disclose generation of a stub.

However, Blumson further discloses: *wherein said instrumented stub is generated from a description of an interface of said second software component* (page 5 paragraph 1; also page 6 Section 6.1 as cited above).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to generate a stub from a description of an interface. One of ordinary skill would have been motivated to generate a stub according to standard practices of using the Open Software Foundation's Distributed Computing Environment (DCE) based on a remote procedure call (RPC) paradigm.

As per claim 3, the above rejection of claim 1 is incorporated. Kazi further discloses: *wherein said second software component is remote from said first software component* (page 8 paragraph 3 as cited above).

As per claim 4, the above rejection of claim 1 is incorporated. Kazi further discloses: *wherein said first software component resides on a first processing device and said second software component resides on a second processing device* (page 8 paragraph 3 as cited above).

As per claim 5, the above rejection of claim 1 is incorporated. Kazi further discloses: *the preliminary step of selecting a log data contents to be included in said stub start and stub end log data, with the selecting step logging zero or more of an application semantic behavior data, a timing latency data, a shared resource usage data, and a causality relationship data* (page 4 paragraph 3).

As per claim 6, the above rejection of claim 1 is incorporated. Kazi does not expressly disclose configuration during generation of the stub.

However, Blumson teaches: *wherein a log data contents is configured during generation of said instrumented stub* (page 6 Section 6.1 paragraph 1: Flags and subflags).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Blumson's flags for stub generation in Kazi's instrumented jvm. One of ordinary skill would have been motivated to minimize perturbation of a target system by inserting only the necessary amount of instrumentation code, thereby reducing the time spent executing logging procedures.

As per claim 7, the above rejection of claim 1 is incorporated. Kazi further discloses: *wherein a log data contents is configured during operation of said component-based software system* (page 5 paragraph 5: "Visualizer").

As per claim 9, the above rejection of claim 1 is incorporated. Kazi further discloses:

initiating said invocation of said second software component from within an execution of a skeleton (page 8 paragraph 3 as cited above);
recording a skeleton start log data before said skeleton invokes said second software component (page 7 last paragraph as cited above); and

recording a skeleton end log data in said skeleton after a response is received from said invocation of said second software component (page 7 last paragraph as cited above).

Kazi does not expressly disclose an instrumented skeleton.

However, Blumson teaches instrumenting server “stubs” or skeletons to collect runtime data (page 6, Section 6.1).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Blumson’s skeleton instrumentation implementation in Kazi’s instrumented jvm. One of ordinary skill would have been motivated to take measurements on certain operations such as marshalling time that are otherwise difficult, while maintaining a relatively simple implementation versus modification of a runtime library.

As per claims 10 and 11, the above rejection of claim 9 is incorporated. All further limitations have been addressed in the above rejections of claims 2 and 3, respectively.

As per claim 13, the above rejection of claim 9 is incorporated. Kazi further discloses: *wherein an accumulated log data from a plurality of instrumented stubs and a plurality of instrumented skeletons is collected and correlated* (Page 3 under “The JaViz performance visualization tool set”).

As per claim 14, the above rejection of claim 9 is incorporated. Kazi further discloses: *wherein said stub start, stub end, skeleton start, and skeleton end log data capture a causality relationship data between said first software component and said second software component* (Page 3 under “The JaViz performance visualization tool set”).

As per claim 15, the above rejection of claim 9 is incorporated. Kazi further discloses: *wherein said stub start, stub end, skeleton start, and skeleton end log data are used to determine a causality relationship data for a plurality of threads* (Page 3, first bullet).

As per claim 16, the above rejection of claim 9 is incorporated. Kazi further discloses: *wherein said stub start, stub end, skeleton start, and skeleton end log data are used to determine a causality relationship data for a plurality of threads spawned during invocation of said second software component* (Page 3 first bullet).

As per claim 17, the above rejection of claim 9 is incorporated. Kazi further discloses: *wherein said stub start, stub end, skeleton start, and skeleton end log data are used to determine a causality relationship data for a thread in which said first software component is invoked* (Page 3 first bullet).

As per claim 18, the above rejection of claim 9 is incorporated. All further limitations have been addressed in the above rejection of claim 5.

As per claim 19, the above rejection of claim 9 is incorporated. Kazi further discloses: *wherein the method includes a transportation of at least a portion of said stub start log data of said instrumented stub to said instrumented skeleton* (Bottom of page 8, e.g. “unique identifier”).

As per claim 21, the above rejection of claim 9 is incorporated. Kazi further discloses: *wherein said instrumented skeleton stores at least a portion of said skeleton start log data to a thread-specific storage* (page 4 third paragraph: “Invocations of the same method executed under different threads are distinguished from one another by their **unique thread identifiers.**”).

As per claim 22, the above rejection of claim 21 is incorporated. Kazi further discloses: *wherein an event number included in said at least a portion of said skeleton start log data is updated before being copied into said thread-specific storage* (Page 9 paragraph 3 “time stamp”).

As per claim 23, the above rejection of claim 9 is incorporated. Kazi further discloses: *retrieving a thread-transportable log data from a thread-specific storage of a*

parent thread; transporting said thread-transportable log data to a child thread; adding a thread information about a child thread to said thread-transportable log data to form a child thread data; and recording said child thread data to a thread table of said child thread (page 12 paragraph 2).

As per claim 24, the above rejection of claim 23 is incorporated. Kazi further discloses: *wherein said thread-transportable log data comprises a self thread identifier and optionally a function container identifier* (page 4 paragraph 3), *<and> distinguishing user-application generated threads from threads generated by an underlying component-based system runtime infrastructure* (page 8 paragraph 2 describes filtering calls from members of the *Java* or *Sun* packages which indicates that the caller is a Java library method and distinguishes it from the distributed monitoring system.).

Kazi does not expressly disclose a self thread identifier that distinguishes user-application threads from component-based threads.

However, it would have been obvious to one of ordinary skill in the art at the time the invention was made to use Kazi's method filter with the thread identifier to produce a unique thread identifier that includes a user/component method designation. One of ordinary skill would have been motivated to produce a complete listing of method calls while enabling filtering to organize results.

As per claim 25, the above rejection of claim 9 is incorporated. Kazi discloses plans to implement memory management debugging (top of page 2), it does not expressly disclose logging heap memory usage data.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to implement memory management logging. One of ordinary skill would have been motivated to debug the garbage collection routines that slow down large applications.

As per claim 26, the above rejection of claim 9 is incorporated. Kazi further discloses: *wherein a particular log data is recorded in a per-process log table* (page 5 paragraph 3).

As per claim 27, the above rejection of claim 9 is incorporated. Kazi further discloses: *wherein a particular log data is recorded on a per-thread basis* (page 7 last paragraph).

As per claim 28, the above rejection of claim 9 is incorporated. Kazi further discloses: *wherein a particular log data is stored in a persistent storage* (page 7 last paragraph).

As per claim 29, Kazi discloses: *processing an accumulated log data and calculating a system behavior characteristic for one or more software components*

executing within said component-based software system (page 5 paragraph 2: “The tree generation step **analyzes the merged trace files to create an output file containing the dynamic execution tree** for a given client or server program. This output file is used by the visualizer to display the call graph.”). All further limitations have been addressed in the above rejections of claims 1 and 9.

As per claims 30 and 31, the above rejection of claim 29 is incorporated. All further limitations have been addressed in the above rejection of claims 14, and 1, respectively.

As per claim 32, the above rejection of claim 29 is incorporated. Kazi further discloses: *wherein said system behavior characteristic comprises a shared resource usage data* (Table 1, “Machine Name”).

As per claim 35, the above rejection of claim 29 is incorporated. Kazi further discloses: *wherein said system behavior characteristic comprises a timing latency data* (Table 1: “Total time”).

As per claim 36, Kazi discloses a computer system (Figure 3). All further limitations have been addressed in the above rejection of claim 1.

As per claim 37, the above rejection of claim 36 is incorporated. Kazi further discloses: *a memory capable of storing said stub start and end log data* (page 8 paragraph 2).

As per claims 38 and 39, the above rejection of claim 36 is incorporated. All further limitations have been addressed in the above rejections of claims 9 and 1, respectively.

As per claim 40, the above rejection of claim 36 is incorporated. All further limitations have been addressed in the above rejections of claims 1 and 4.

As per claim 41, the above rejection of claim 36 is incorporated. Kazi further discloses: *wherein said memory further includes a thread table adapted to store thread log data* (page 7 last paragraph).

As per claim 42, the above rejection of claim 36 is incorporated. All further limitations have been addressed in the above rejection of claim 28.

8. Claims 8, 12, and 43 are rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Kazi and Blumson as applied to claim 7, 9, and 36, respectively above, and further in view of U.S. Patent 5,522,073 to Courant et al. (hereinafter referred to as "Courant").

As per claim 8, the above rejection of claim 7 is incorporated. Kazi does not expressly disclose regular expressions.

However, in an analogous environment, Courant teaches using a customizable regular expression to limit the information (column 8 lines 26-32).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Courant's regular expressions in Kazi's visualization system. One of ordinary skill would have been motivated to select only the messages that are of interest in order to clearly display results and maximize log utility.

As per claim 12, the above rejection of claim 9 is incorporated. Kazi does not expressly disclose enabling and disabling logging.

However, Courant teaches enabling a logging operation by sending messages (column 11 lines 23-31).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Courant's teaching of sending messages to enable logging in Kazi's distributed monitoring system. One of ordinary skill would have been motivated to control logging capabilities to minimize system perturbation when logging is not needed.

As per claim 43, the above rejection of claim 36 is incorporated. All further limitations have been addressed in the above rejections of claims 28, 29, and 12.

9. Claim 20 rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Kazi and Blumson as applied to claim 9 above, and further in view of U.S. Patent 5,146,593 to Brandle et al. (hereinafter referred to as "Brandle").

As per claim 20, the above rejection of claim 9 is incorporated. Kazi does not expressly disclose the use of additional parameters in data transportation.

However, in an analogous environment, Brandle teaches the use of additional parameters to pass information to a function defined in an interface definition (column 4 line 67 – column 5 line 6).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Brandle's teaching of parameters to transport information in Kazi's distributed monitor. One of ordinary skill would have been motivated to provide caller information that would permit the callee to more quickly perform a task. By providing information as a parameter, the callee would not need to further analyze a function call since the information would be readily available.

10. Claims 33 and 34 are rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Kazi and Blumson as applied to claim 29 above, and further in view of "Unix Power Tools" by Peek et al. (hereinafter referred to as "Peek") .

As per claims 33 and 34, the above rejection of claim 29 is incorporated. Kazi does not expressly disclose CPU usage data.

However, in an analogous environment, Peek teaches the use of the Unix "ps" command which provides statistics regarding CPU and memory usage.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Peek's teaching of CPU usage in Kazi's distributed monitor. One of ordinary skill would have been motivated to analyze the relative use of system resources in an effort to locate performance bottlenecks.

Conclusion

11. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Particularly, IBM Technical Disclosure "Statistics Gathering and Analyzing Tool for Open Software Foundation's Distributed Computing Environment" discloses instrumentation of client/server "stubs" and keeping logs regarding execution.

12. Any inquiry concerning this communication or earlier communications from the examiner should be directed to J. Derek Rutten whose telephone number is (703) 605-5233. The examiner can normally be reached on M-F 6:30-3:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (703) 305-4552. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

jdr


WEI Y. ZHEN
PRIMARY PATENT EXAMINER